**Document Manager**

Document manager is a class with static methods in utils package, which is in charge of the CRUD operations on documents.

Method **\_\_add\_upload\_document** takes as arguments a document name, author which is the currently logged user, abstract short description of the document, keywords which could be used for searching by the most important words in the document, status which could be Draft, Final, Final Revised or Blocked. This function also receives a file which is the actual file to be uploaded. The result of the method execution is that a new document with the arguments properties is stored.

Method **\_\_add\_rector\_disposition\_document** takes as arguments a document name, the author of the document, abstract short description of the document, keywords, status and the user initiating the request. This method creates a new rector disposition document with attributes initialized corresponding to the arguments received in method signature. A new docx type document is created from template and the newly created “.docx” document is save.

Method **\_\_add\_necessity\_request\_document** takes as arguments the document name, the author, the abstract description of the document, keywords, status and currently logged in user. A new Necessity request document object is created and the arguments are set on the object’s fields. A new “.xlsx” document is created from the template and saved.

Method **\_\_update\_upload\_document** takes as arguments the database id of the file, the new document name, the new abstract description of the document, the new keywords, the new status and the new selected file to be uploaded. The old instance of the document is retrieved from the database with the specified id, then fields document name, abstract description, keywords and the file are updated on the retrieved object from the database. The final step before updating the document instance in the database is to update the version number.

Method **\_\_update\_rector\_disposition\_document** takes as arguments an id, a new document name, new keywords, new status, and a context, which is used to generate file content. The disposition document is retrieved from the database querying it by it’s id which is received in the method arguments. The disposition document fields are updated with the fields received in method arguments. The file path is saved, the version of the document is upgraded and then it is saved in the database.

Method **\_\_update\_necessity\_request\_document** takes as arguments an id, a new document name, new keywords, new status, new context from which the content of the file will be generated. The necessity request document having the id given in the parameters is retrieved from the database, it’s fields are updated with the values given as parameters to the function. A new ‘.xlsx” file is generated with new context content, the file is saved on the disk, the version is upgraded and then saved in the database.

Generic method **add\_document,** receives a type, a document name, an author, abstract description, keywords and status. This function is in charge of deciding which add document method should be called based on the type with the received parameters. Type could be **Upload**, **DR** (disposition rector), **RN** (request necessity).

Generic method **update\_document** receives an id, a type, a document name, an abstract description, keywords and a status. This function is in charge of deciding which update document method should be called based on the type with the received parameters. Type could be **Upload**, **DR** (disposition rector), **RN** (request necessity).

Method **remove\_document** takes an id and queries the database for the document having the id equal with the received id and deletes it’s physical file by it’s path and the entry corresponding to the document is also deleted from the database.

Method **make\_doc** received a context, based on which it generates a “.docx” file, saves it and returns it’s file path.